

TrustNoExe – A executable filter for Windows 2000

It's now a common daily occurrence to receive PE viruses via e-mail. On Windows platforms, nine out of ten of last year's top viruses were spread via e-mail. While staff training is the best deterrent, wouldn't it be helpful to prevent users opening un-trusted executables yet being non-restrictive on the opening documents and other less harmful files?

With typical figures saying 70% of network related attacks come from within your organisation doesn't it make sense to prevent users running port scanners or other executable tools from floppy disk or CDROM drives yet still allowing the use of these drives to transfer files and maintain office efficiency.

Or perhaps you have caught users trying to install software on machines. While operating systems are becoming more secure, it is still possible to install programs as a user or run programs directly from the CDROM drive. Other users may choose to play games from CDROM drives?

The speed at which viruses can propagate must be a concern for all Administrators. Most sites now have automatic updates running which frequently update their scanners, sometimes as frequently as twice daily. However it takes time after a virus is released, to first be detected and identified and then to be added to the virus definitions of all the major virus scanners before the site administrator even gets their hand on it. Many people will remember the SQL Slammer Worm. Its peak occurred only three minutes after it was released to the wild. At its peak it was scanning 55 million Internet hosts per second and had infected at least 75,000 victims.

If any of these issues are of a concern, you need trust-no-exe.

Doesn't Windows NT/2000/XP have an execute permission?

Out of the box, Windows NT/2000 and XP together with a NTFS file system will provide the administrator with an execute permission per file. This permission is shared with the transverse folder right and can be used to prevent an executable from loading while still allowing it to be read or written to. However when applied to an executable the user receives the rather bland message "Access to the specified device, path, or file is denied". This can confuse the user into thinking an error has occurred, rather than the fact they are not permitted to execute this program.

However the biggest disadvantage to this scheme is the administrator has no control over drives which do not have a NTFS file system or compatible network file system. Drives such as a 1.44MB 3.5" Floppy, CDROM, DVD, Zip Drive or even some network drives do not have adequate security descriptors and thus cannot be adequately secured. Removing or disabling these drives is one option but doing so greatly effects the productivity that should be gained from a PC Workstation.

What is Trust-No-Exe?

Trust-no-exe, simply put, is an executable file filter. It attaches to the operating system and filters all executable files, be it .exe .com .dll .drv .sys .dpl etc from all drives and all network shares against a list of files or paths, you, the administrator provide as trusted applications. If a prohibited executable (one not in the allow list or one defined in the deny list) is loaded, a popup box informs the user with an intelligent message. This message can be customised to your site including a contact or phone number reinforcing to the user that this is not an error.

On the other hand perhaps you are not ready for total lock down just yet, but are worried about all these PE viruses, executable christmas/birthday cards, screen savers etc that are coming in via email. While most of your users don't click on these you are worried about security holes in your email client, either hiding extensions or embedding files into html messages.

By placing the email attachment directory in the denied list and removing all entries from the allow list you can prevent users (or malicious scripts) from running any executable attachments from the default directory. However the user can still move the file to different location to run it just like they can run all other programs outside of this explicitly defined directory. The popup message box can be customised to remind users that it is company policy not to open executable files received by email. But what happens if the executable's don't have .exe or hidden extensions? How will trust-no-exe know if they are executable or data files?

Trust-no-exe hooks into the operating systems routines for creating a process and loading it into memory. If the operating system attempts to load any compiled code into memory ready to give it execution as a process or thread, trust-no-exe will jump on it and prevent the code from being loaded into memory. Therefore trust-no-one doesn't rely on the file extension.

Other uses of the deny list can be adding "`c:\programs files\outlook express`" to prevent Outlook or the Windows Address book from loading. Or you could add "`c:\winnt\system32\wscript.exe`" to prevent .js scripts to run. On the other hand D:\ and A:\ could be added to prevent users loading programs from Floppy and CDROM drives yet still access and copy files from these drives.

Trust-no-exe has been designed for ease of use. Out of the box, a control panel applet is installed which allows for the configuration to be modified. By default the program files and winnt/windows directories are added which in many cases is all that is required to make a secured, yet functional system. You can then change the permissions on these folders to prevent write access and thus prevent your users from adding any additional un-trusted executables to your portfolio or trusted programs.

New in version 3 is the ability to add a custom message. This allows you to put in a contact name and number should your users require special access to certain files. The other unique feature of Trust-no-exe is the file denied dialog is a single executable that is called by the trust-no-exe driver. The program spawned can be found at %systemroot%\system32\trustnoexe\denyexe.exe. Therefore you can create your own dialog with company logo should the need arise. Please contact us if you would like to explore this option. We can assist my providing a Visual C++.net or Borland C++ Builder Template.

It is just as important, if not more is to have trust-no-exe protection when logged in as an Administrator. Trust-No-Exe protects your PC all the time regardless of what user is logged in. To install software, or run executables from un-trusted locations, the administrator can utilise the control panel to stop the driver to briefly interrupt filtering while the software is installed. Trust-No-Exe also protects tasks running in the SYSTEM account.

How it works

TrustNoExe consists of three components –

- A monolithic kernel mode driver loaded at boot time that acts as the filter.
- A control panel applet that is used to configure the driver specifying the trusted applications at your site.
- A substitute executable file displaying a message indicating to the user that the program they just opened is prohibited.

The driver attaches itself to the operating system's Create Section function that is used to load executable code into memory during the creation of a process. In plain english, it attaches itself to a part of the operating system that handles the loading of all compiled code whether it is a .exe, .com, .sys, .dll, .scr, .cpl, .api, .drv, .bpl or other executable objects.

Emphasis should be put on *compiled*, as it is still possible to interpret high level code such as visual basic scripts or java applets etc. For example clicking on a .js (java script) file spawns the Windows Based Script Host, wscript.exe. wscript will then read the javascript file as text and interpret the contents. In this case if you wanted to deny access to javascript files you can add the following line to the deny list "`c:\winnt\system32\wscript.exe`".

Every time the CreateSection function is called, trust-no-exe makes a check to see if the file being called is allowed to be loaded into memory. If so, it allows the loading and hence the execution of the file. On the otherhand if the file is not allowed, trust-no-one replaces the handle to the prohibited file with that of a "You are not allowed to execute this program" message executable. This reduces any interaction between usermode and kernel mode not only making the program more efficient but also allows custom applets to be written and called by the site administrator.

While this could simply be an error message with your organisation's logo and/or policy on it, it could also extend to logging either locally, or via email etc. This is completely adaptable by the administrator to suit the needs of your site.

Installing Trust-No-Exe

Installation of trust-no-exe is easy. Simply run install.exe from the trust-no-exe distribution. After accepting the licence agreement trust-no-exe is installed on your system. At completion of the install, the trust-no-exe driver is not immediately started. It will start on the next reboot or can be manually started by using the trust-no-exe control panel applet.

This gives the administrator time after the install to verify that your filtered paths are correct. Paths can be added or deleted by using the control panel applet, which was installed during installation. When installed for the first time, the executables in c:\winnt (for Windows NT and 2000), c:\windows (for Windows XP) and c:\program files are allowed. Take caution when setting the paths as removing c:\winnt or c:\windows can prevent your computer from functioning correctly.

Installing Trust-No-Exe on Multiple Workstations

Installing software and modify settings on multiple computers is never fun, yet alone efficient. Cloning computer images using software such as Norton Ghost is one option. However with the introduction of Trust-no-exe version 3, you need only install the package on a single workstation using the standard installation procedure above. Once installed and appropriately configured, you may utilise the Multiple Workstation functionality to remotely install it with your configuration on other selected computers. All that is needed is a single click.

Likewise changes to the access list can be distributed almost instantly and with minimal fuss. Computer groups compatible with beyondexec can be quickly loaded, or you can effortlessly create your own using the built in computer picker.

To start installing your Trust-No-Exe setup on other workstations, open up the Trust-No-Exe control panel applet and click on the Multiple Computers button. This expands the dialog to allow you to enter a list of computers to target. Once your list of computers is specified, you can click on the Apply Settings button. This will try to connect to each computer in the list and install, configure and start trust-no-exe over the network with a minimal of fuss. If it cannot log into the remote computer with the current credentials, it will prompt for a username and password. This user must have administrative privileges on the remote machine.

After the initial installation, if you have the requirement to modify your access lists, simply pressing Apply Settings will copy your new settings to the list of specified computers. Should at a later date, you want to remove trust-no-exe and it's restrictions, you can click on the remove all button. This will remove all registry keys, device drivers and control panel applets on the remote group of computers, leaving you without a trace of Trust-No-Exe. Life could not be simpler . . .

Precautions

. . However, rapidly distributing software to multiple computers at once can also cause multiple failures at once. While Trust-No-Exe has been written with as much protection in mind, it is wise to test the procedure on one or two computers before doing 10, more, or your entire organisation. Trust-No-Exe allows you to create, load and save .grp (group) files. These are compatible with beyondexec and are simply a plain text file with the computer names on each line. Therefore you may want to group up your PC's into groups of 10 to 20 computers each and install each group independently.

Care should also be taken when crossing between or mixing Windows NT/2000 and XP machines. Windows NT/2000 share the same \winnt\ directory whereas Windows XP uses the \windows\ directory. While this does not effect the primary installation, as the Trust-No-Exe installer is intelligent enough to obtain the system path, it does effect your access list you port to other machines. If you do intend to work across platforms, it may be wise to add both \windows and \winnt to your access list. Otherwise set up two installations, one for Windows NT and 2000, and another one for Windows XP.

Troubleshooting & FAQ

I have a network drive mapped to h:\. However when I add h:\ to the allow list, I still can't execute my programs on h:\ - How does trust-no-exe handle network drives?

As users can un-map and re-map drives with their restricted rights, Trust-no-exe converts all network drive paths to UNC paths. If you intend to allow files to be loaded from network drives the UNC path should be used. e.g. if I:\ was mapped to \\mars\temp then to allow the execution I:\hello.exe would require \\mars\temp\hello.exe to be added to your allow list.

I have a folder which requires write rights and contains a trusted executable. However I don't want to allow users to run other applications that they may place in this folder. How do I set up Trust-No Exe?

Trust-No-Exe will first check the filename/path against the allow list. If a match is made on the full path name and executable (i.e the executable is explicitly defined) it will immediately grant access to the executable. However if a match is only made on the path, it will then check the deny list for a matching entry. If one exists, the program is not allowed to run, otherwise if no match is made in the deny list, the program is allowed to run.

Lets take for example we have Microsoft Office installed in the c:\program files\microsoft office\office\ directory. However office requires write access to this directory so users could potentially place files in this folder and execute them. We want to deny access to any executable in this folder except for winword.exe, powerpnt.exe, excel.exe and osa.exe. We also have other programs installed in c:\program files\ which we want to use.

Access List

```
c:\winnt
c:\program files\
c:\program files\microsoft office\office\winword.exe
c:\program files\microsoft office\office\excel.exe
c:\program files\microsoft office\office\powerpnt.exe
c:\program files\microsoft office\office\osa.exe
```

Deny List

```
c:\program files\microsoft office\office\
```

If we set up Trust-No-Exe with the above settings, any programs in c:\program files\ will be granted temporary access by the allow list. It is then checked against the deny list and provided the programs are not in c:\program files\microsoft office\office\, they will be executed. On the other hand if they are in this folder, they will be denied execution. However if we run winword.exe for example, it is explicitly defined in the allow list and hence will be granted immediate execution access without checking the file against the deny list.

You will then want to set the permissions on winword.exe, excel.exe, powerpnt.exe and osa.exe so users cannot overwrite them with other programs.

My files in c:\program files doesn't work!

Some older 16 bits programs require entries for c:\program files and c:\progra~1

I can't install Trust-No-Exe remotely on a Windows XP Home computer.

Remote registry RPC calls required by Trust-No-Exe is not available in the Windows XP Home Edition. If you want this functionality, please upgrade to Windows XP Professional. Trust-No-Exe will however function fine if installed locally using install.exe found in the Trust-No-Exe distribution.

When I use the control panel applet to install Trust-No-Exe on Windows XP Professional Computers in a Workgroup, I get "Access Denied. Please log in as an Administrator."

Please ensure the "Network access : Sharing and security model for local accounts" is set to "Classic – local user authenticates as themselves" This can be found in the Local Security Settings, Local Policies. Note that

this policy is enabled to “*Guest only – local users authenticate as guest*” by default for a computer running Windows XP Professional that is joined to a workgroup.

Can I remove the Trust-No-Exe Control Panel Applet? I don't want users to be able to change the settings.

A user without administrative rights can open the Trust-No-Exe control panel applet. While with restricted rights, they can view the access lists they *cannot* modify them, nor can they start, stop or display the status of the driver.

Will Trust-No-Exe work on Windows 95, Windows 98 or Windows ME?

No, trust no exe requires certain system calls that are only available on NT kernels found in NT/2000 or XP. There is no intention to support these operating systems. If security is a concern, we would recommend moving to Windows XP.

Manually uninstalling

Don't ask me how I did it, but my computer no longer boots up correctly – perhaps it is trust-no-exe?

If you believe the trust-no-exe driver is preventing critical executables from loading due to a mis-configuration of the filter paths don't despair. Boot to the Windows NT/2000/XP recovery console so you have access to the file system. Delete the driver, c:\winnt\system32\drivers\bltrust.sys.

Now re-boot your system. Windows may mention it cannot load a driver, just ignore this for the moment. This annoying dialog box can be removed by deleting the following registry key :
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\bltrust or by using the Trust-No-Exe un-install wizard found in control panel, add/remove programs.

I have a feature that would be useful to me. How do I go about getting it implemented?

Any features you would like to see implemented can be e-mailed to trust-no-exe@beyondlogic.org

Trust-No-Exe Revision History

- 1st August 2003 - Version 3.0 Windows NT/2000/XP.
 - Rewritten Control Panel Applet to reduce code size. Added updated dialogs, Custom Messages and Multiple Workstation Support. Control Panel Applet now unloads and uninstalls properly.
 - Rewritten denyexe.exe to accept custom messages. Reduced code size.
 - Rewritten GetDriveDeviceObject routine. No longer seeks the A: drive and improved the speed and reliability of the routine.
 - Modified HookZwCreateSection routines to improve reliability across WinNT/2000 and XP.
 - Fixed problem with running programs on network shares mounted on Windows XP.
 - Modified installer so driver is automatically loaded by the service control manager and not by the system loader.
 - Modified driver so if registry keys are missing, driver is inactive. This prevents the computer from failing to boot in rare instances.
 - Tested on Windows 2000 SP3 & SP4, Windows NT4 SP5 & SP6(a), and Windows XP & SP2.
- 3rd October 2002 - Version 2.1 (Free demo) Windows NT/2000/XP.
 - Tested on Windows 2000 SP2 & SP3, Windows NT4 SP5 & SP6(a), and Windows XP.
- 1st October 2002 - Version 2.0.
 - Added support for Windows NT and Windows XP. Driver now automatically detects the O/S and hooks the appropriate O/S specific functions.
- 27 June 2002 - Version 0.9 (Beta/demo) Windows 2000 Only.
 - First demo release for public evaluation
- 13 April 2002 - Version 0.2 (Release Candidate).
 - Added installation program and control panel applet. Tested only on Win2000.
- 20 February 2002 - Version 0.1.
 - Proof of Concept.